CS670: Assignment 1

Instructions

- Using LaTeX to typeset your solutions will fetch +5 bonus marks in the assignment.
- You can ask the instructor for help in the assignment during office hours or via appointment.
- All help will stop 48 hours before the assignment deadline.
- **Deadline: March 4th 2024, EOD**

1 Applications of DPFs

There are two servers $\mathcal{S}_0$ and $\mathcal{S}_1$. There are C clients. Each client holds a string $\alpha_i \in \{0,1\}^n$. The servers want to learn the number of clients who hold a certain string σ. At the same time clients would like to keep their strings a secret.

1. Write a protocol that allows the servers to calculate the number of clients who hold σ (while learning nothing about the client's secret string) **(12 marks)**.
2. Prove the correctness of your protocol **(3 marks)**.

2 Authenticated PIR

Observe that Private Information Retrieval (PIR) does not guarantee the integrity of the database, and thus it does not guarantee the integrity of the response that a client receives. Suppose that an honest database holder does not have enough space to hold the database. Therefore, it outsources the storage to an untrusted party. A client now uses PIR to retrieve data from this untrusted party.

Malicious Database Holder

The malicious (and untrusted) *single* database holder can alter some contents of the database. Suggest a way in which you could prevent the malicious database holder from altering the database contents. Write protocol in detail. **(5 marks)**

A Two Server PIR protocol

Now let us consider two servers who hold the replicas of the database.

1. Write a simple protocol that allows a client to privately retrieve the $\mathtt{i}^{\text{th}}$ record of the database. **(3 marks)**
2. Now, assume that one of the servers is malicious and can tamper with the database. Would the above protocol still work? Why/Why not? **(2 marks)**
3. Modify the above protocol so that we can detect any foul play from the malicious database holder. *(Hint: Along with the shares of a standard basis vector the client sends the shares of a standard-basis vector scaled with a random α.)*. Prove the correctness. **(10 marks)**

3 Computational PIR

Replication is needed

Prove that if there are no *computational assumptions*, a single server is not enough to achieve PIR. The proof should be detailed and formal. **(5 marks)**

SPIR

A Symmetric PIR scheme (henceforth SPIR) is a PIR scheme where, at the end, Alice learns nothing more than x_i. We will allow the databases to share a common random string; however, the length of that string will be one of our parameters. (Recall that a is Quadratic Residue modulo m, if $z^2 \equiv a \pmod{m}$) **Hint:** User wanting to download item i, downloads $n-1$ random quadratic residues modulo m, $a_1, a_2, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n$. User also generates one quadratic non-residue b_i. The user then sends $a_1, a_2, \ldots, a_{i-1}, b_i, a_{i+1}, \ldots, a_n$. Server then computes $R \leftarrow u_1^{X_1} \cdot u_2^{X_2} \cdot \ldots \cdot u_n^{X_n}$. Where $(X_1, X_2, \ldots, X_n)$ is a one bit database held by the server and u_i's are the series of the numbers received by the server. Of course, recall that server cannot distinguish between a residue and a non-residue.

1. Describe the CPIR protocol based on the quadratic residue assumption. Why does the protocol work? **(5 marks)**
2. Is the Quadratic Residue based PIR protocol SPIR? Prove it either way. **(10 marks)**

4 Programming Distributed Point Functions

1. Clone the repository from `git@github.com:avadapal/cs670-iitk-2024.git`. **(2 marks)**
2. Use the Makefile to compile the code and run the code. **(3 marks)**
3. The current code does not give the expected output. What should be the expected output? **(5 marks)**
4. Make changes to the `traverse` function in `dpf.h`, so that we get the expected output. **(10 marks)**